REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)* 03 May 2004	**2. REPORT TYPE** Technical Paper/Slides	**3. DATES COVERED** *(From - To)*

4. TITLE AND SUBTITLE
Web-Based Telemetry Station Software

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Washburn, W. Gregory

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) AND ADDRESS(ES)
JT3 LLC
190 S. Wolfe Ave., Bldg. 1260
Edwards AFB, CA 93524

8. PERFORMING ORGANIZATION REPORT NUMBER
PA-04108

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
412 TW/DRP
Air Force Flight Test Center (AFFTC)
Edwards AFB, CA 93524

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
N/A

12. DISTRIBUTION / AVAILABILITY STATEMENT
A Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES
CC: 012100 CA: Air Force Flight Test Center Edwards AFB

14. ABSTRACT

The Advanced Range Telemetry Integration and Support (ARTM I&S) program is developing a novel web-based system for automated testing and setup of telemetry receiving systems. This system shares test equipment between multiple tracking systems and allows for automated bit error rate (BER) as well as solar calibration tests. Complete receiver and bit synchronizer configurations can be stored to and recalled from a central database. The web-based implementation easily allows remote control utilization. This paper describes the software architecture and capabilities.

15. SUBJECT TERMS
Web-based remote control; automated testing

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON** W. G. Washburn
a. REPORT UNCLASSIFIED	**b. ABSTRACT** UNCLASSIFIED	**c. THIS PAGE** UNCLASSIFIED	Unclassified Unlimited	9	**19b. TELEPHONE NUMBER** *(include area code)* 661-277-9010

Web-Based Telemetry Station Software

W. Gregory Washburn
JT3, LLC
190 S Wolfe Ave, Bldg 1260, Div 1522
Edwards AFB, CA 93524
Voice: 661-277-8052, Fax: 661-277-2594
E-Mail: william.washburn@edwards.af.mil

I. Abstract

The Advanced Range Telemetry Integration and Support (ARTM I&S) group is developing a novel web-based system for automated testing and setup of telemetry receiving systems. This system shares test equipment between multiple tracking systems and allows for automated bit error rate (BER) as well as solar calibration tests. Complete receiver and bit synchronizer configurations can be stored to and recalled from a central database. The web-based implementation easily allows remote control utilization. This paper describes the software architecture and capabilities.

II. Introduction

Edwards AFB is upgrading their telemetry infrastructure and rewriting the control software for automated setup and testing.

The concept of operations is that there is an operator for each antenna and a single analyst that sets up and tests each system and acts as an "extra set of eyes." The analyst controls the communications analyzer and the signal generator that feed the boresite antenna for BER tests used for all of the antennae. A central database of the receiver configurations was desired to facilitate initial setup and moving missions over antennae during a mission.

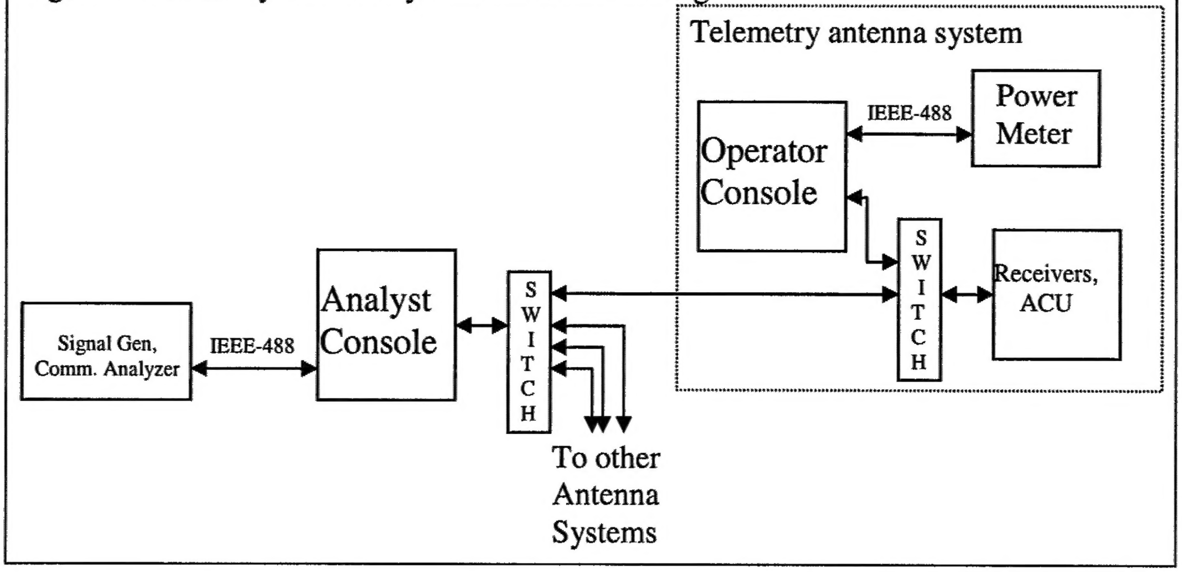
Sharing test equipment over several systems, the ability to have two or more persons checking configurations, and having a central database lends itself to a distributed, web-based control system. The development system chosen was the Microsoft.Net environment since it is web-ready and the development system of the future.

For security reasons, this system will not be connected to the Internet or the Edwards Intranet.

III. Hardware Configuration

The hardware configuration is shown in Figure 1. The analyst console is networked to each of the antenna systems. Using Ethernet switches rather than hubs reduces network traffic.

Figure 1. Telemetry Control System Hardware Configuration

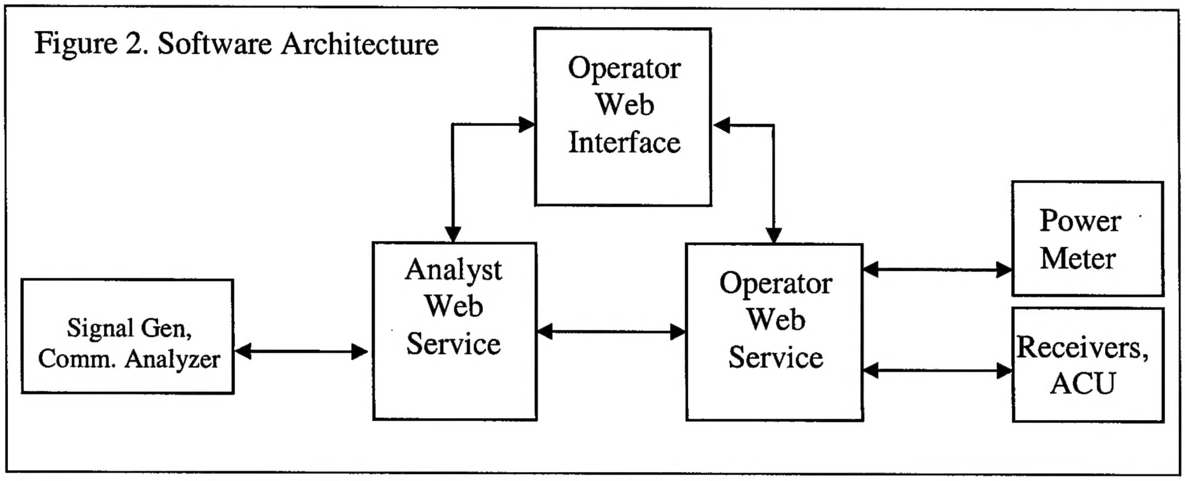


The receivers being used are the L-3 RCB-2000s. They are controlled over Ethernet and come with remote control software. L-3 modified their firmware to allow for multiple simultaneous Ethernet connections, which permits the operator and the analyst to see the receivers' statuses simultaneously and lets the web-based software communicate with the receivers for automated tests or setup.

IV. Software Architecture

Since the control of the equipment is distributed, the software is broken up into three smaller programs: operator interface, operator web service, and analyst web service. Figure 2 shows how these components communicate with each other and with external equipment.

Figure 2. Software Architecture



The operator web interface was written in ASP.Net/VB.Net. Both web servers were written in ASP.Net/C#.Net. All programs are object-oriented with objects written for each piece of external equipment. These objects contain the remote commands specific to that equipment. If equipment is swapped out for another model or manufacturer, only that object would need to be replaced and the program recompiled.

A. Operator Web Service

Figure 3 shows the software architecture for the operator web service. The program uses the StreamReader class to read a text file containing the addresses of all of the connected equipment. This allows use of the same software for all of the systems just by changing the configuration file. This file also contains the antenna coordinates for calculating the sun position (G/T), antenna pointing angles to the boresite antenna for doing BERs, and antenna number to display on the operator interface.

One of the challenges to this project was to find a GPIB/IEEE 488 card that had drivers written for the .Net environment. Although the .Net software was released more than 2 years ago, only two manufacturers were found that support .Net: Contec and National Instruments. We were able to borrow and test a National Instruments card, so this is the card we purchased.

Figure 3. Operator Web Service

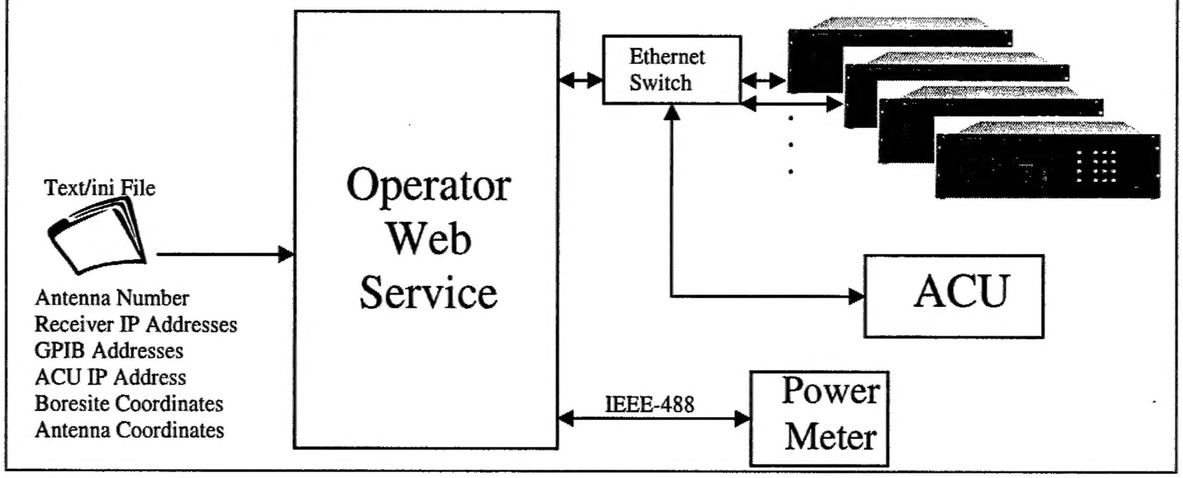


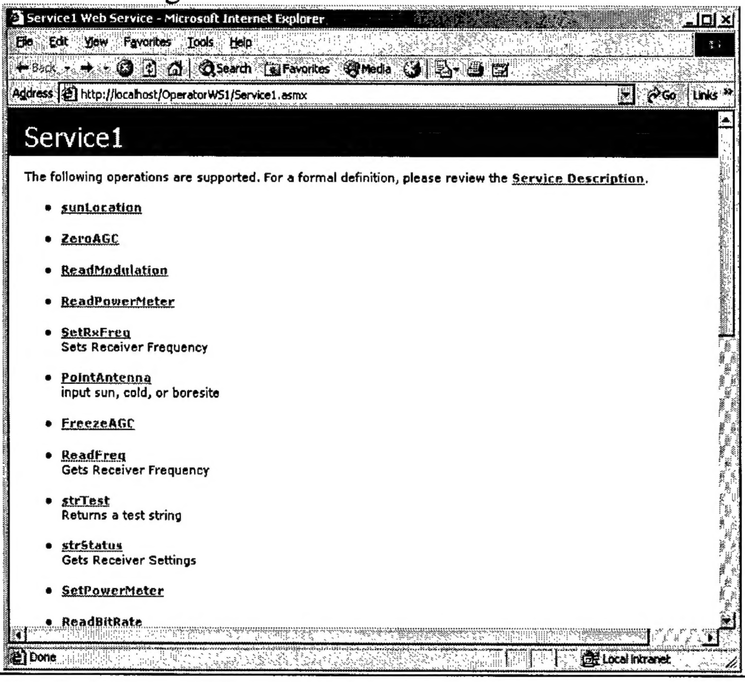
Figure 4 shows how one exposes a web service method. One must make it public by declaring it as a WebMethod.

Figure 4. Web Service Method

```
[WebMethod(Description="Sets Receiver Frequency")]
public string SetRxFreq(int RCB, string strFreq)
{
      GetIPAddress(RCB);
      sSend(Receiver.rSetRxFreq() + strFreq);
      vShutdown();
      return "ACK";
}
```

Figure 5 shows the web service page while running. This is handy when troubleshooting the code.

Figure 5.Web Service Page


Service1 Web Service - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://localhost/OperatorWS1/Service1.asmx
Go Links
Done
Local intranet

Service1

The following operations are supported. For a formal definition, please review the Service Description.

- sunLocation
- ZeroAGC
- ReadModulation
- ReadPowerMeter
- SetRxFreq
  Sets Receiver Frequency
- PointAntenna
  input sun, cold, or boresite
- FreezeAGC
- ReadFreq
  Gets Receiver Frequency
- strTest
  Returns a test string
- strStatus
  Gets Receiver Settings
- SetPowerMeter
- ReadBitRate

Clicking an item brings up another window that asks for the parameters to pass, if applicable.

A socket is created to communicate over Ethernet. A sample of the code is shown below in Figure 6:

Figure 6. Ethernet socket excerpt

```
//In the Declarations section
Socket sock = new Socket(AddressFamily.InterNetwork,SocketType.Stream,
ProtocolType.Tcp);
.
.
.
//In the parsing section
//str[RCB] is an array of strings with the receiver addresses
System.Net.IPAddress addr = System.Net.IPAddress.Parse(str[RCB]);
System.Net.IPEndPoint ipEP = new System.Net.IPEndPoint(addr,port);
.
.
.
private string strEthRead()
{
        string resp = "";
        try
                {
                // Create an IpEndPoint to capture the identity of the
                // sending host.
                IPEndPoint sender = new IPEndPoint(IPAddress.Any, 0);
        EndPoint tempRemoteEP = (EndPoint)sender;

        // Create a byte buffer to receive the message.
        byte[] buffer = new byte[1024];

        // Receives datagram from a remote host.
        sock.ReceiveFrom(buffer, ref tempRemoteEP);

        resp = System.Text.Encoding.ASCII.GetString(buffer);
        }//end of try
    catch(Exception e)
        {
        resp = "Exception : " + e.ToString();
        }
    return resp;
}//end of strEthRead
```

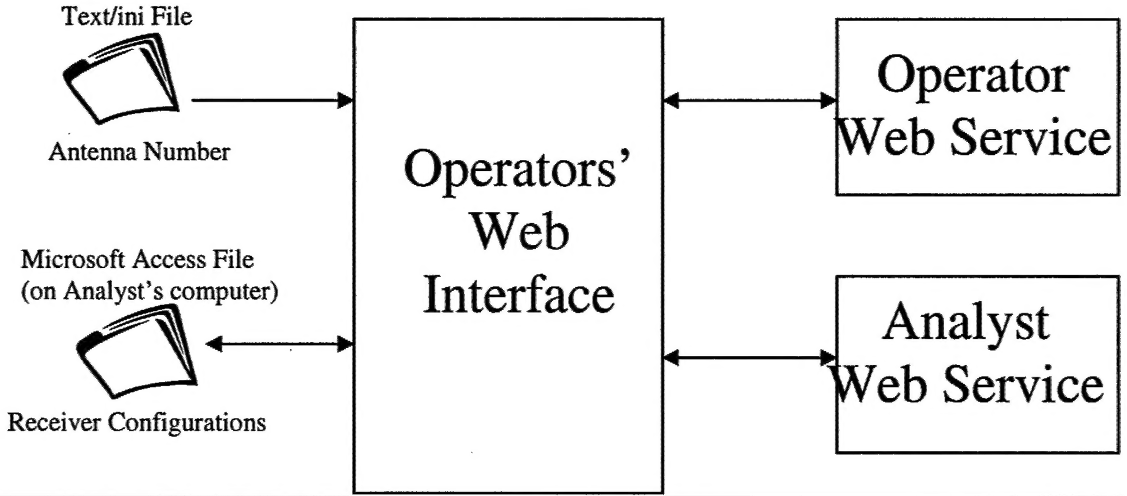
The sample above is written in C# and shows that the language is not intuitive.

National Instruments provides programming samples for using their cards. Adapting their examples to work from a web service was not difficult.

B. Operator Web Interface

The operator web interface, Figure 7, is made up of two separate web pages. One page is for automated tests, the other for receiver configuration.

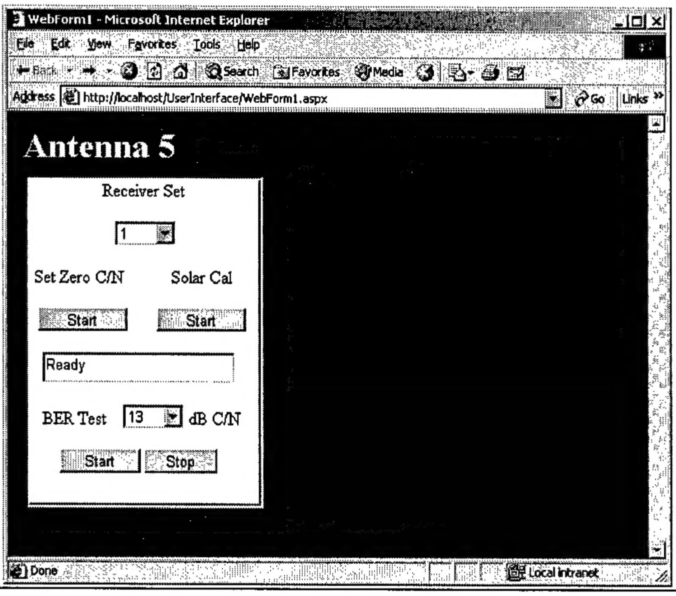
Figure 7. Operator Web Interface



The page for automated tests communicates through proxies to the test equipment using the respective web services. To consume a web service, one needs to add a web reference to the Visual Studio .NET project, then declare an instance of the proxy and use it as any other class.

The user can zero the receivers and run automated BER and solar calibration tests from the test screen, Figure 8.

Figure 8. Operator Test Screen


WebForm1 - Microsoft Internet Explorer

Address http://localhost/UserInterface/WebForm1.aspx

Antenna 5

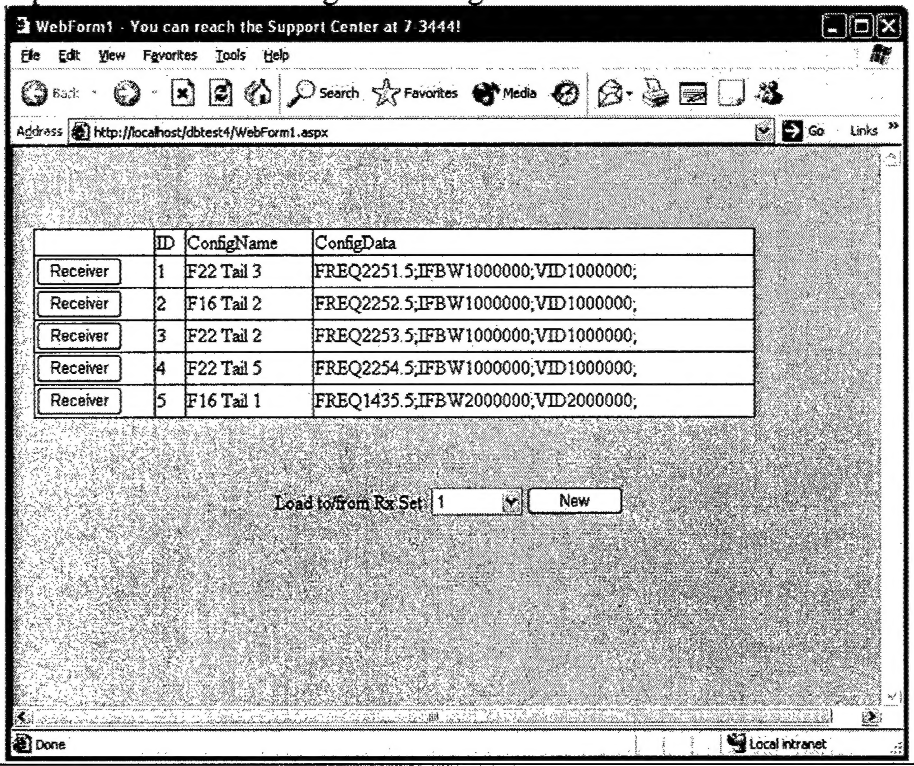
Receiver Set: 1

Ready

BER Test 13 dB C/N

The operator's receiver configuration page is shown in Figure 9. The operator could save a new configuration by first setting up the receivers manually or through the L-3 software, then saving the configuration from this page. The program reads the configuration from the receivers and saves it to the database.
The configuration database resides on the analyst's computer. This allows all of the antennae to access the same database. In the .Net environment, the web page loads the database into an intermediate dataset, and then reads the dataset into the data grid displayed on the page. Updates are made to the dataset, which updates the database.

Figure 9. Operator Receiver Configuration Page



WebForm1 - You can reach the Support Center at 7-3444!
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://localhost/dbtest4/WebForm1.aspx
Go Links
Done
Local intranet

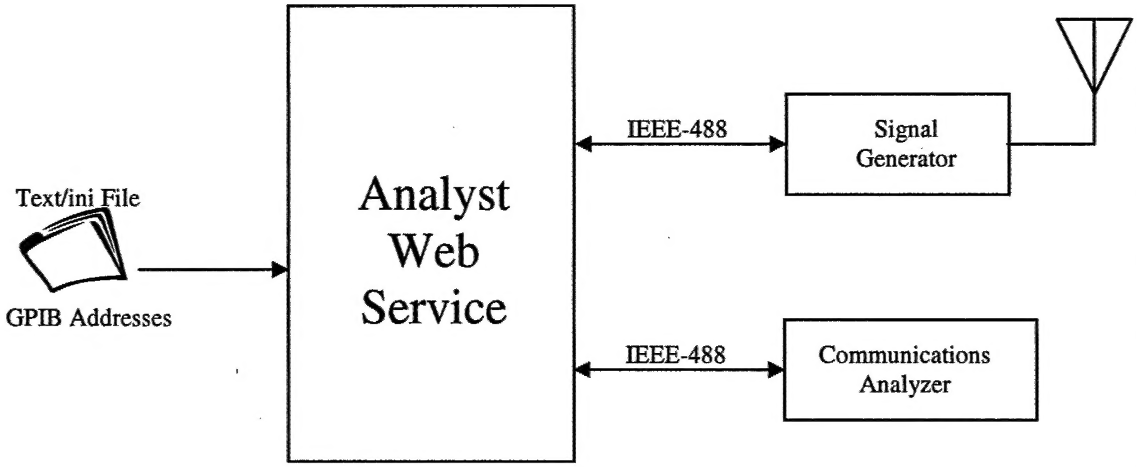
	ID	ConfigName	ConfigData
Receiver	1	F22 Tail 3	FREQ2251.5;IFBW1000000;VID1000000;
Receiver	2	F16 Tail 2	FREQ2252.5;IFBW1000000;VID1000000;
Receiver	3	F22 Tail 2	FREQ2253.5;IFBW1000000;VID1000000;
Receiver	4	F22 Tail 5	FREQ2254.5;IFBW1000000;VID1000000;
Receiver	5	F16 Tail 1	FREQ1435.5;IFBW2000000;VID2000000;

Load to/from Rx Set: 1 New

C. Analyst Web Server

The analyst web server, shown in Figure 10, operates like the operator web service. The equipment addresses are held in a text file and software objects were written for the individual commands to the hardware.

Figure 10. Analyst Web Service



V. Remote Control

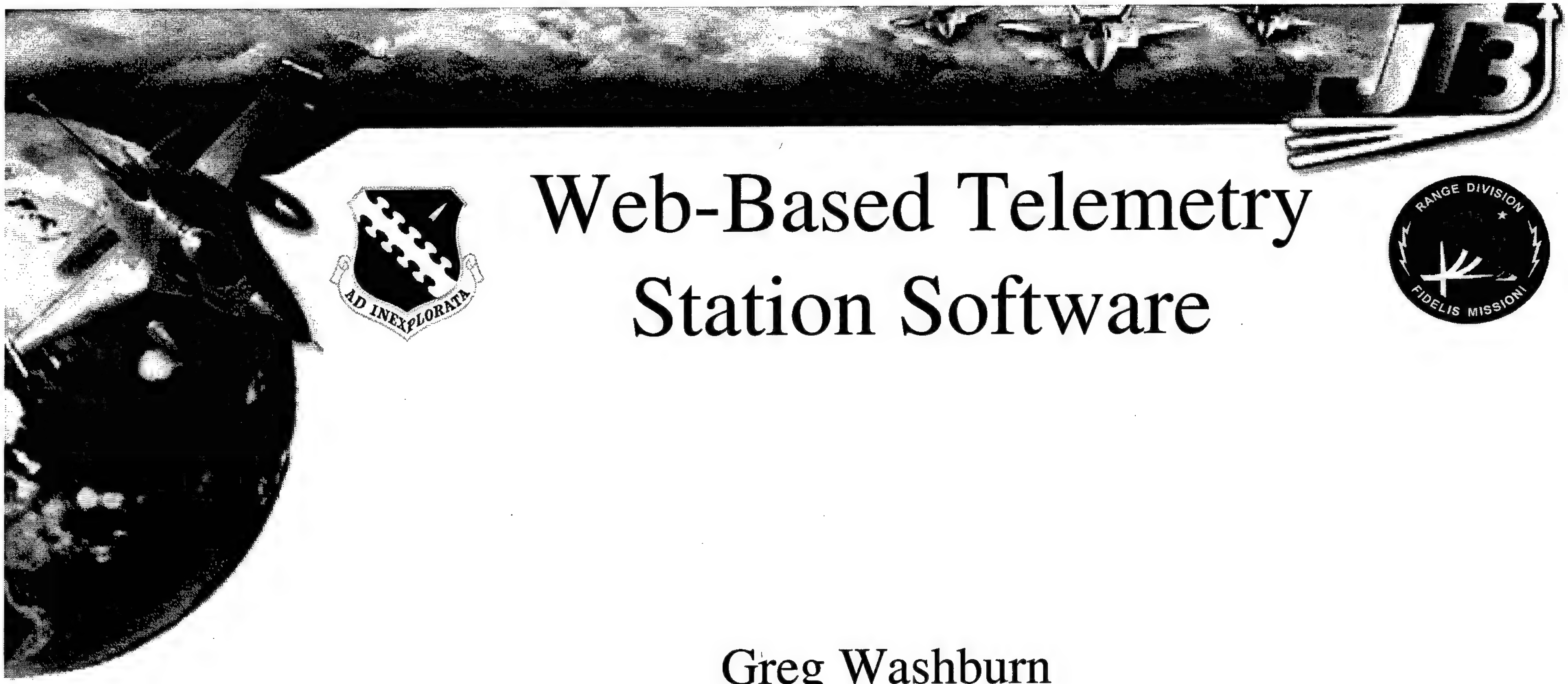
When it becomes the required time to upgrade Edwards AFB's mobile telemetry stations and remote control, this software can be reused. All that would be necessary is a computer with a web browser and an Ethernet connection and there would be the same interface as at a fixed site. Of course, one would have to remote the antenna control unit and other equipment.

VI. Future Enhancements

Future enhancements to this project are to display the oscilloscope, spectrum analyzer, and possibly even boresite camera videos on a web page. This would allow sending these signals to a control room to show reasons for data quality, etc.

VII. Conclusions

There are pros and cons to using web-based control software. On the positive side, one can create an interface as simple or complex as required. Extending control is as easy as connecting an Ethernet cable. Once the software has been loaded additional software is not required on remote computers. Any number of systems may be networked with no additional coding. However, some of the drawbacks are the steeper software development learning curve (.NET) than previous versions of Visual Studio and compatible PCI cards are harder to find.



Web-Based Telemetry Station Software

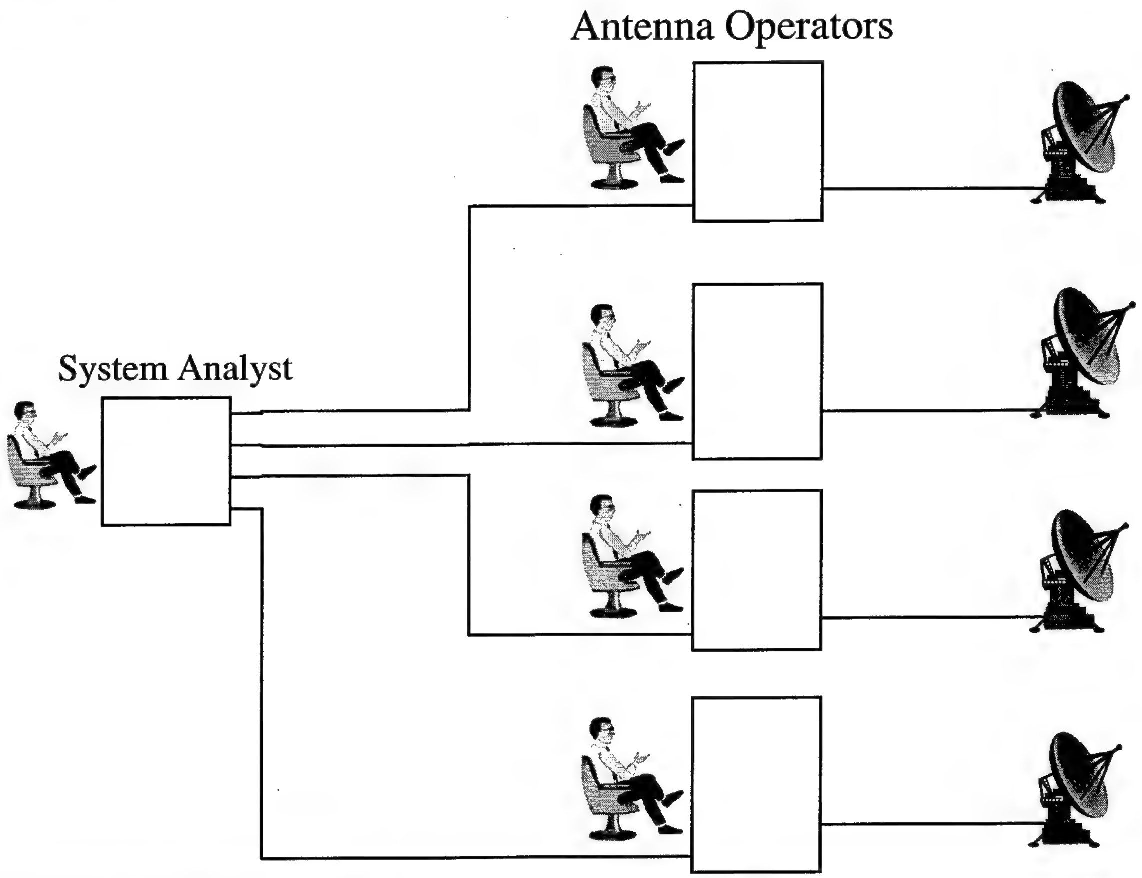
Greg Washburn
JT3 LLC
Edwards AFB CA
661-277-8052

Web-Based Telemetry Station Software

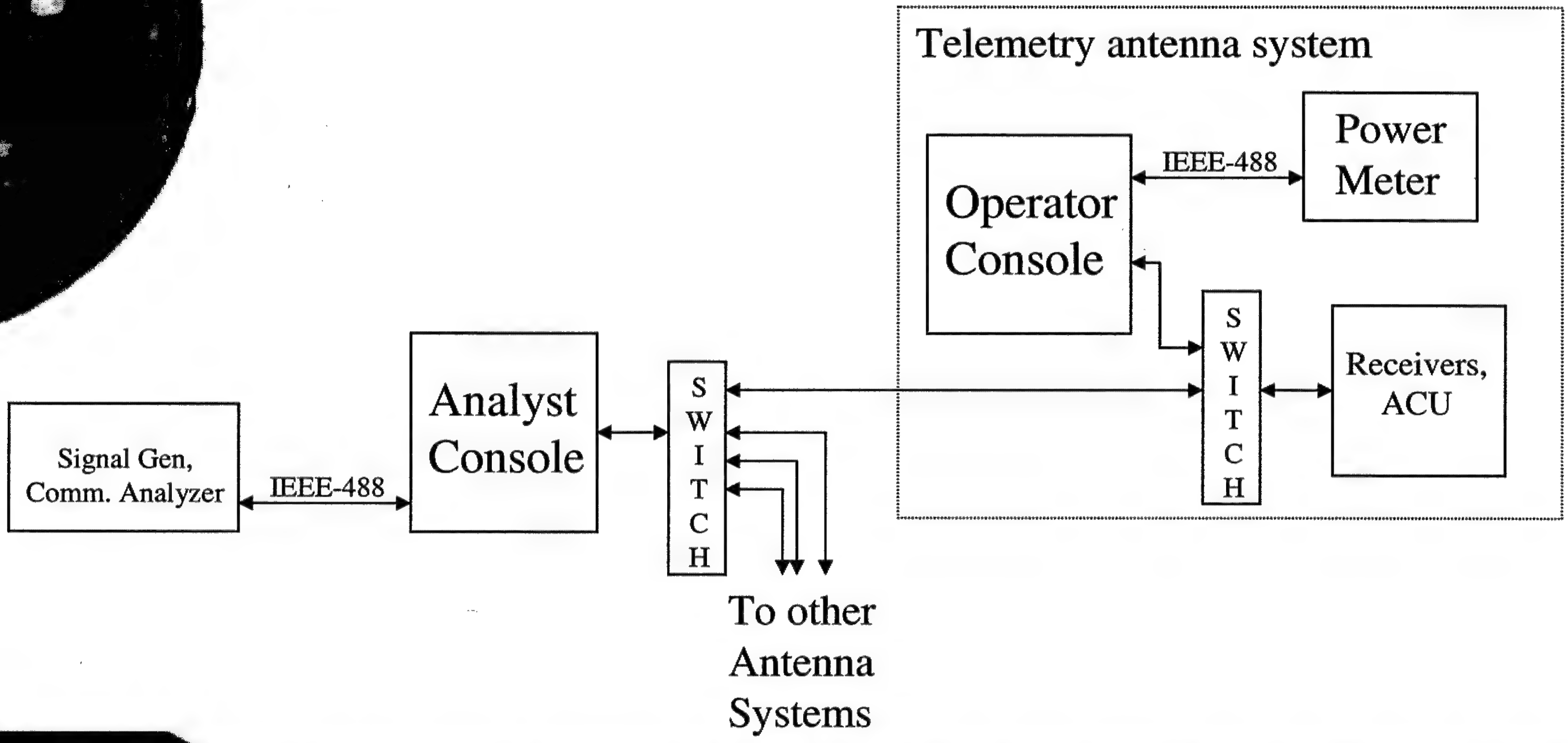
Abstract:

- Advanced Range Telemetry Integration and Support (ARTM I & S) group is developing a web-based system for automated testing and setup of telemetry receiving systems
- Shares test equipment between multiple tracking systems
- Allows for automated bit error rate (BER) and solar calibration tests
- Complete receiver and bit synchronizer configurations can be stored to and recalled from a central database
- Easily allows for remote control
- This paper describes the software architecture and capabilities.

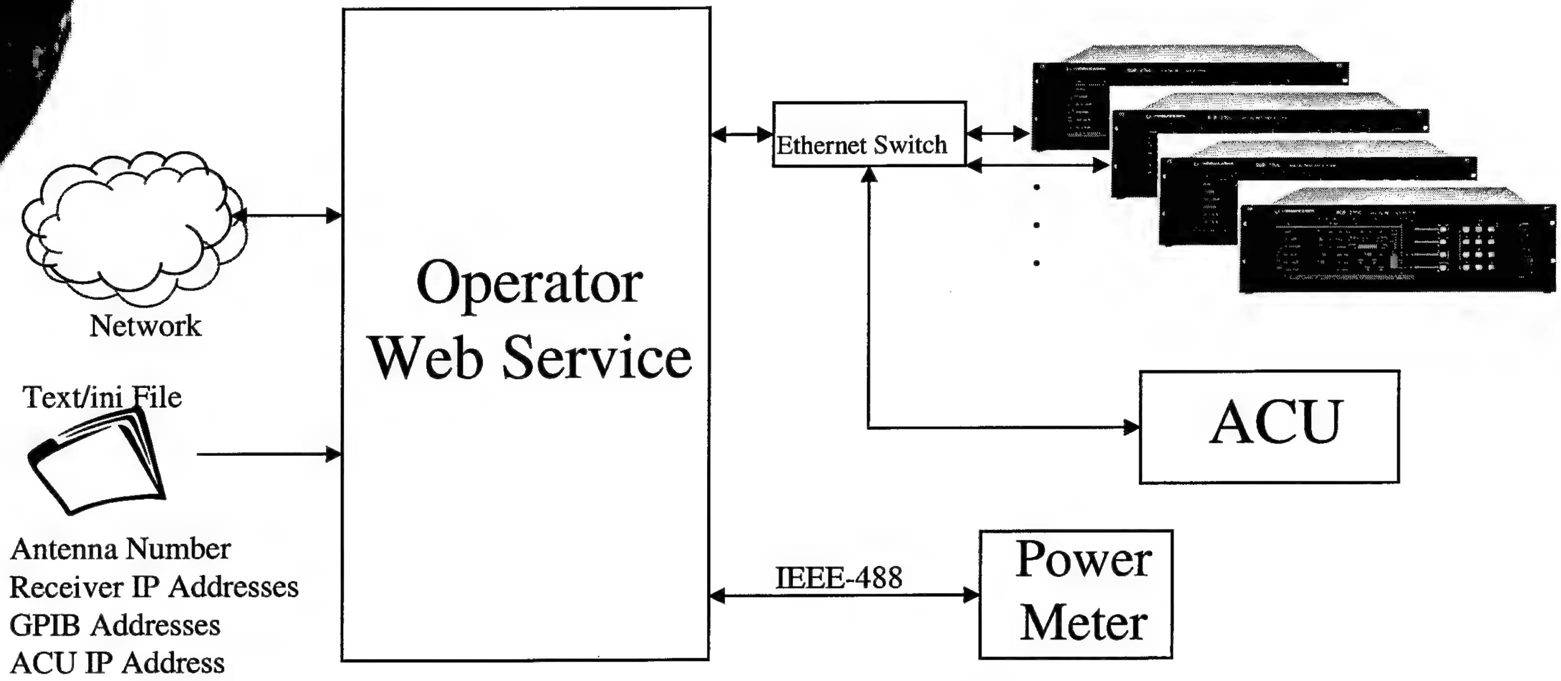
Overall Block Diagram



Telemetry Station Block Diagram



Operator Web Service Block Diagram



Network

Text/ini File

Antenna Number
Receiver IP Addresses
GPIB Addresses
ACU IP Address

Operator Web Service

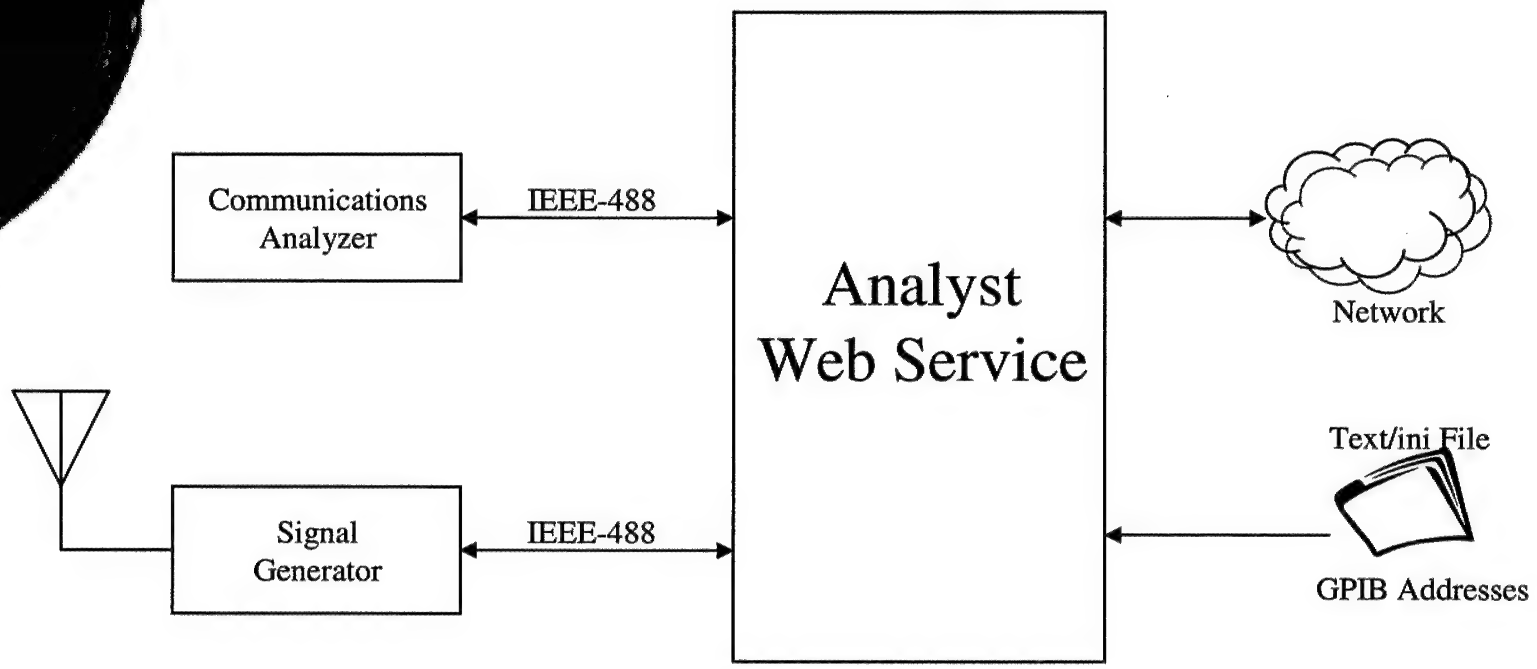
Ethernet Switch

ACU

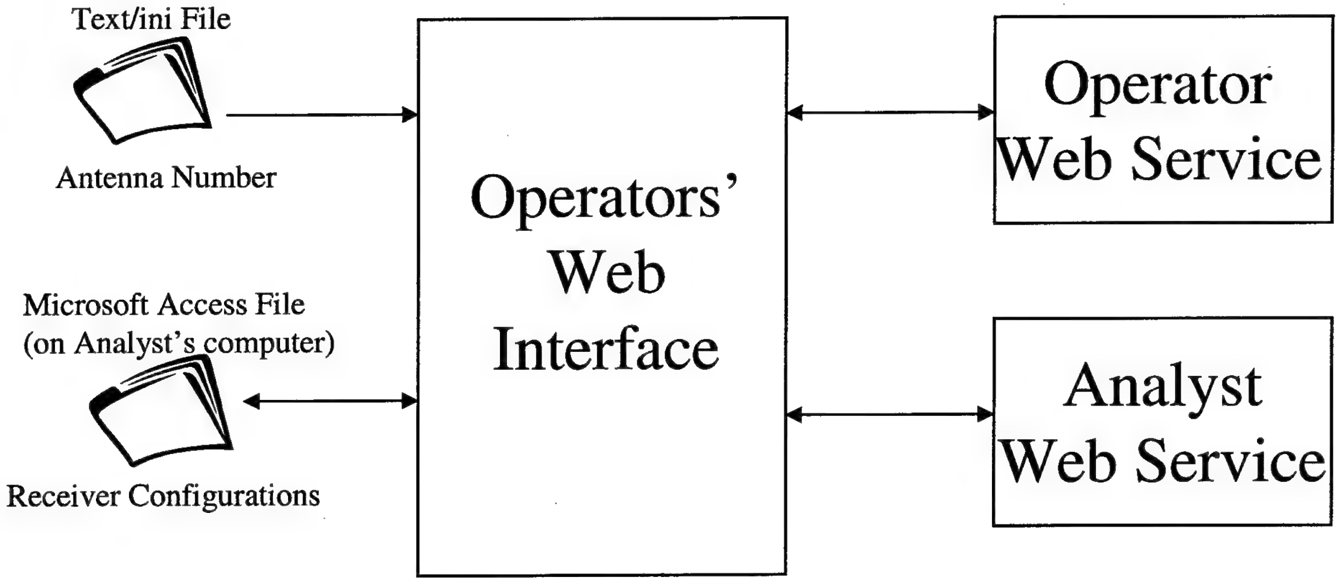
IEEE-488

Power Meter

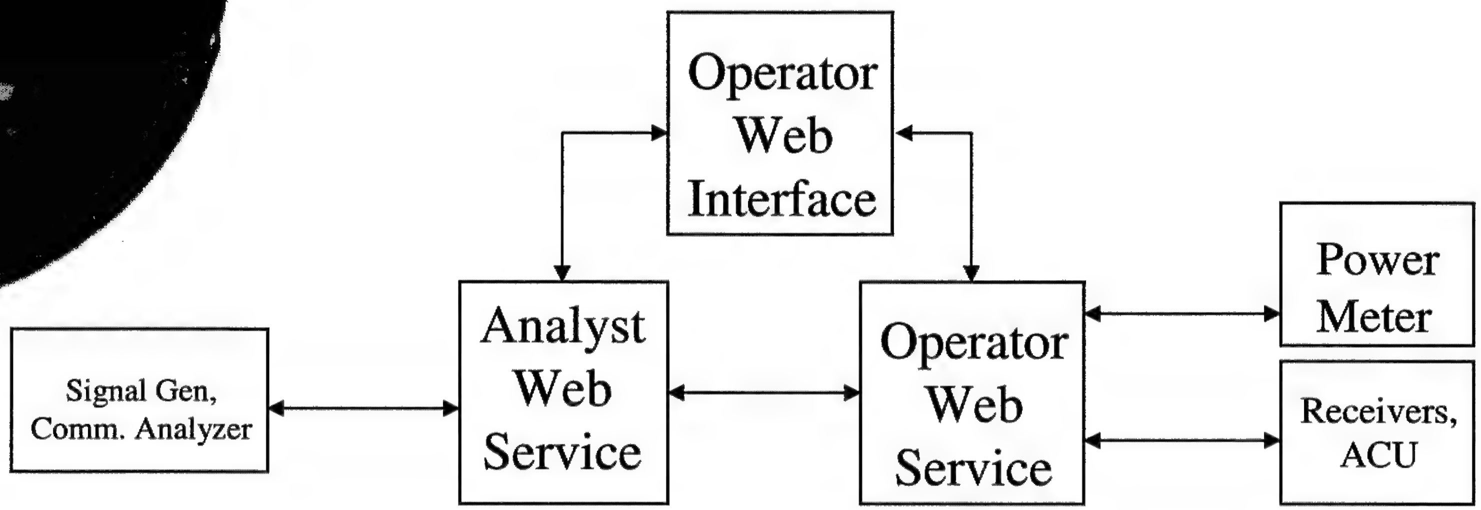
Analyst Web Service Block Diagram



Operator Web Interface



Software Block Diagram



Operator Web Interface

Signal Gen, Comm. Analyzer

Analyst Web Service

Operator Web Service

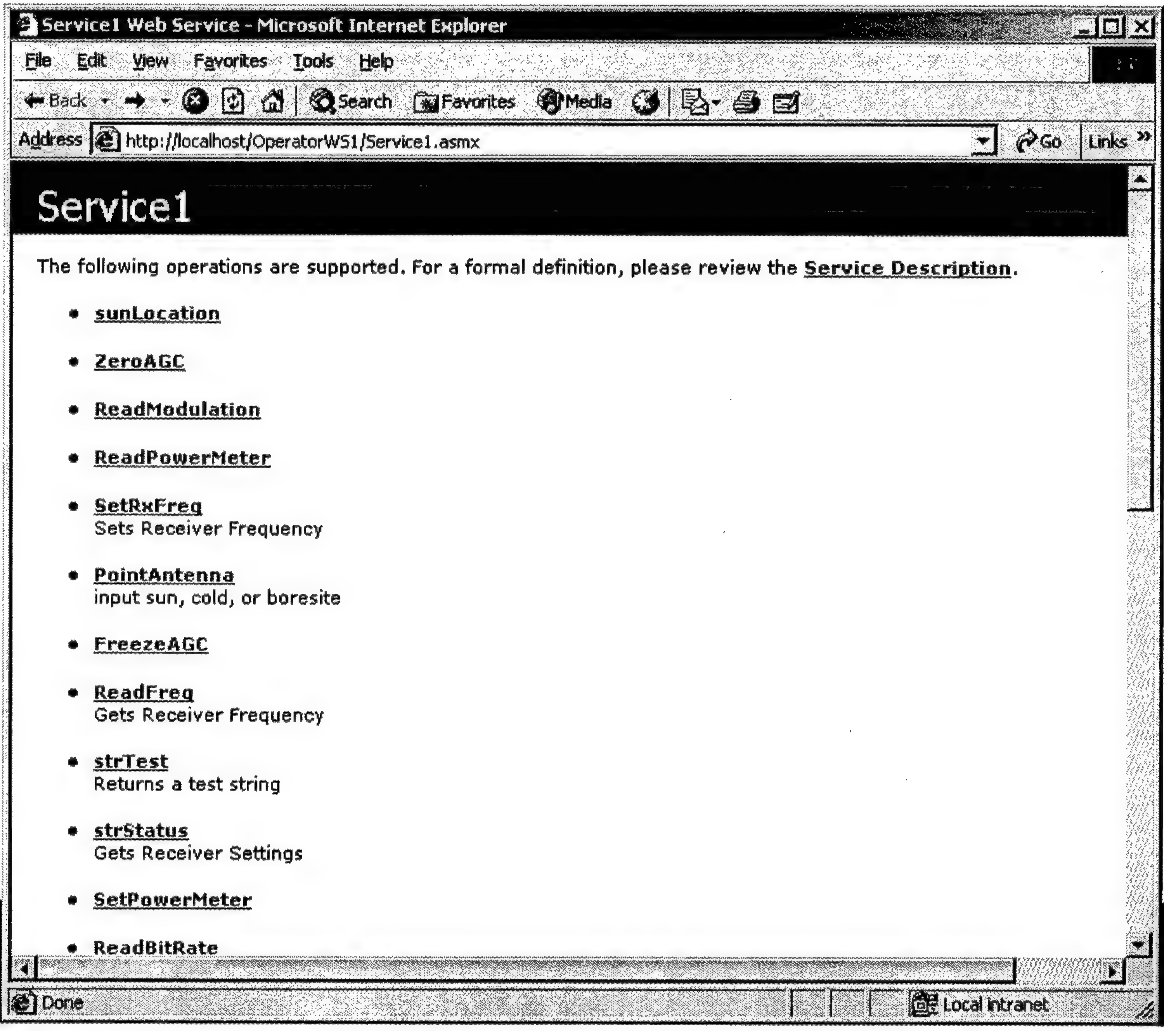
Power Meter

Receivers, ACU

Declaring a Web Service Method

```
    [WebMethod(Description="Sets Receiver
Frequency")]
    public string SetRxFreq(int RCB, string strFreq)
    {
        GetIPAddress(RCB);
        sSend(Receiver.rSetRxFreq() + strFreq);
        vShutdown();
        return "ACK";
        }
```

Web Service Interface

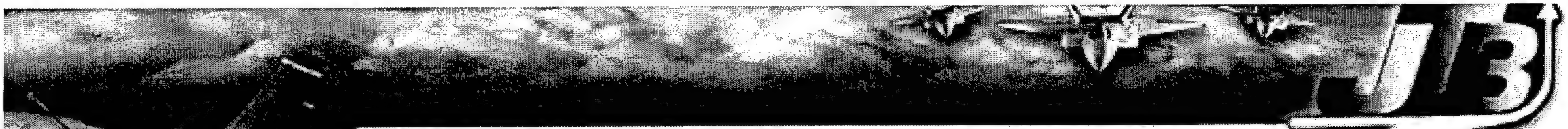

Service1 Web Service - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back
Search
Favorites
Media
Address
Go
Links
Done
Local intranet

http://localhost/OperatorWS1/Service1.asmx

Service1

The following operations are supported. For a formal definition, please review the **Service Description**.

- **sunLocation**
- **ZeroAGC**
- **ReadModulation**
- **ReadPowerMeter**
- **SetRxFreq**
  Sets Receiver Frequency
- **PointAntenna**
  input sun, cold, or boresite
- **FreezeAGC**
- **ReadFreq**
  Gets Receiver Frequency
- **strTest**
  Returns a test string
- **strStatus**
  Gets Receiver Settings
- **SetPowerMeter**
- **ReadBitRate**



Ethernet Read Code Example

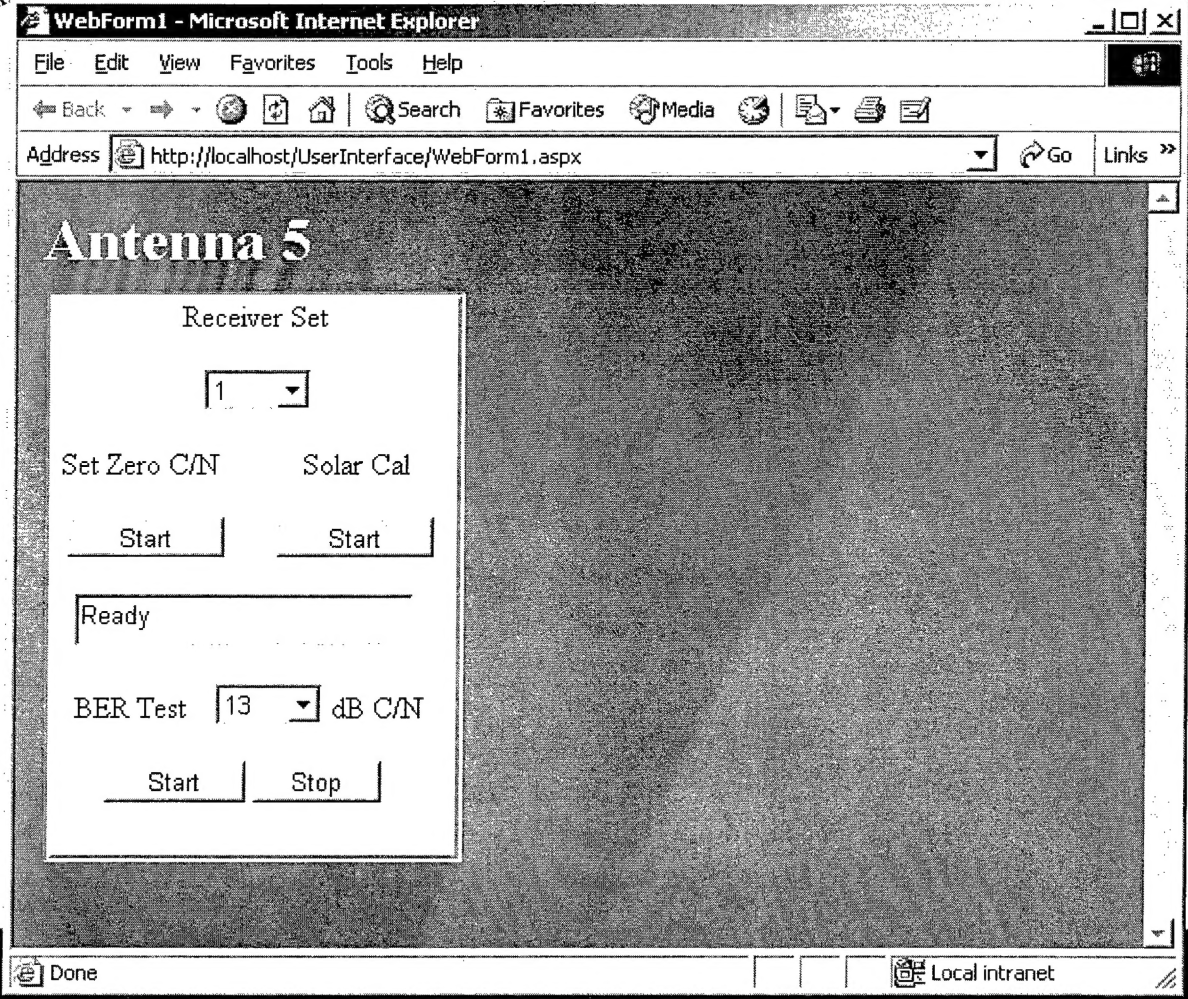
```
private string strEthRead()
{
    string resp = "";
    try
    {
        // Creates an IpEndPoint to capture the identity of the sending host.
        IPEndPoint sender = new IPEndPoint(IPAddress.Any, 0);
        EndPoint tempRemoteEP = (EndPoint)sender;

        // Creates a byte buffer to receive the message.
        byte[] buffer = new byte[1024];

        // Receives datagram from a remote host.
        sock.ReceiveFrom(buffer, ref tempRemoteEP);

        resp = System.Text.Encoding.ASCII.GetString(buffer);
    }//try
    catch(Exception e)
    {
        resp = "Exception : " + e.ToString();
    }
    return resp;
}//strEthRead
```

Operator Interface



WebForm1 - Microsoft Internet Explorer

Address http://localhost/UserInterface/WebForm1.aspx

Antenna 5

Receiver Set 1

Set Zero C/N

Solar Cal

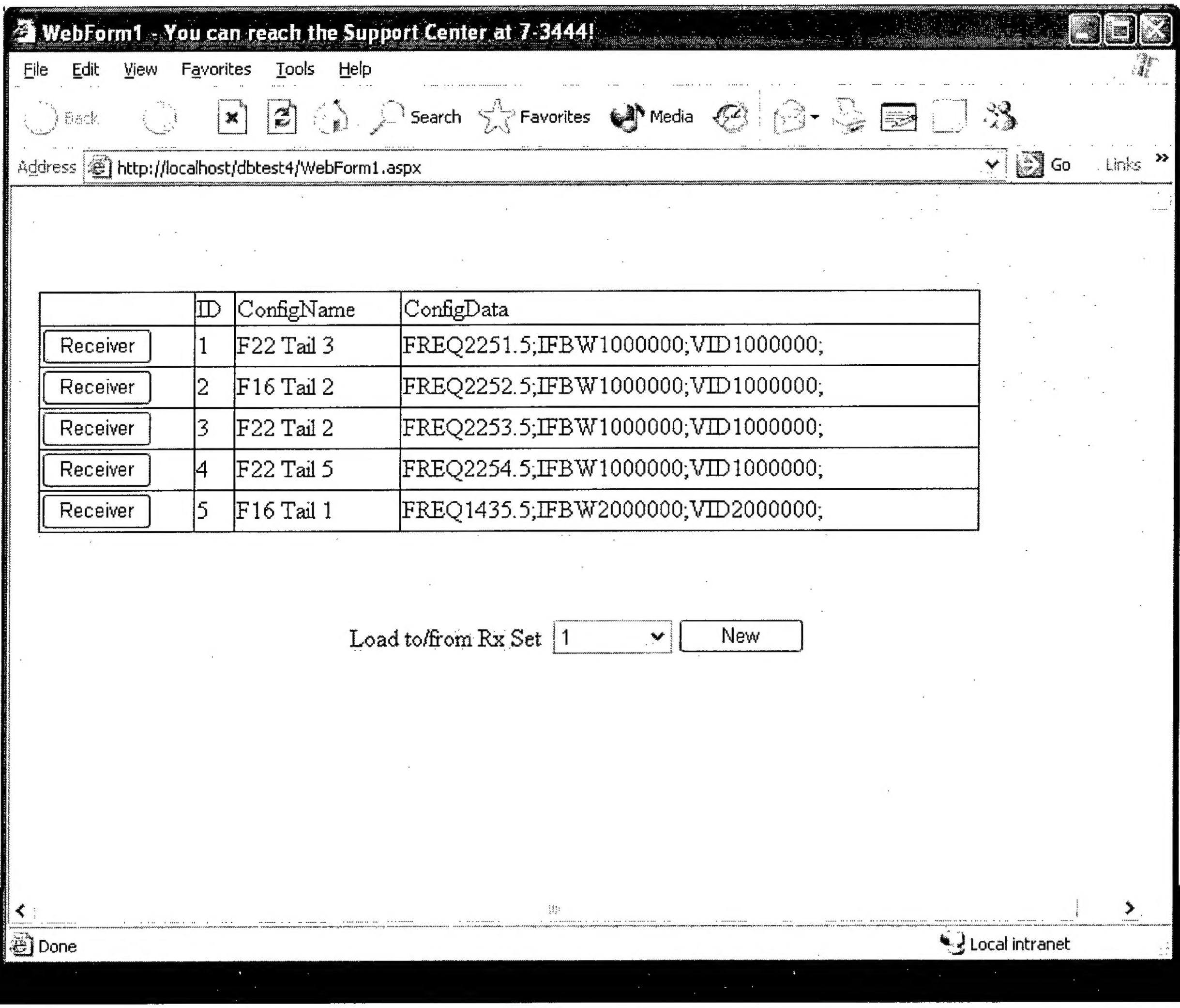
Ready

BER Test 13 dB C/N

Done

Local intranet

Receiver Configuration Page



WebForm1 - You can reach the Support Center at 7-3444!
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://localhost/dbtest4/WebForm1.aspx
Go Links
Done
Local intranet

	ID	ConfigName	ConfigData
Receiver	1	F22 Tail 3	FREQ2251.5;IFBW1000000;VID1000000;
Receiver	2	F16 Tail 2	FREQ2252.5;IFBW1000000;VID1000000;
Receiver	3	F22 Tail 2	FREQ2253.5;IFBW1000000;VID1000000;
Receiver	4	F22 Tail 5	FREQ2254.5;IFBW1000000;VID1000000;
Receiver	5	F16 Tail 1	FREQ1435.5;IFBW2000000;VID2000000;

Load to/from Rx Set 1 New

Future Enhancements

- Display the oscilloscope, spectrum analyzer, and possibly even boresite camera videos on a web page
- Allows sending these signals to a control room to show reasons for data quality, etc.

Conclusions

Using web-based control software has its pros and cons.

Pros:

- One can create an interface as simple or complex as required
- Easy to extend remote control
- Any number of systems can be added with no additional coding

Cons:

- Steeper learning curve on software
- Compatible PCI cards are harder to find